Mobile Voice over IP (MVOIP): An Application-level Protocol for Call Hand-off in Real Time Applications

G. Ayorkor Mills-Tettey and David Kotz
Dartmouth College
ayorkor@alum.dartmouth.org, dfk@cs.dartmouth.edu

Abstract

This paper presents Mobile Voice Over IP, an application-level protocol to support terminal mobility in real-time applications such as voice over IP, on a wireless local area network. We describe our MVOIP implementation based on the ITU-T H.323 protocol stack, present experimental results on call hand-off latency, and discuss various implementation issues, including the task of quickly and accurately determining when call hand-off is necessary. We also discuss how MVOIP relates to other proposed mobility support schemes, and how it can be generalized to provide application-level mobility support in a wide range of real and non real-time applications.

1. Introduction

Voice over Internet Protocol (VOIP) applications transmit real-time data, such as voice or video, over IP networks. Two major protocol stacks for Voice over IP are the H.323 protocol stack [1], by the International Telecommunications Union (ITU), and the Session Initiation Protocol [5], by the Internet Engineering Task Force (IETF). The former is a vertically integrated suite of protocols for voice, video and data communication over packet-based networks, whereas the latter is a more flexible standard for initiating multimedia sessions between endpoints. The differences in the protocols reflect the differing backgrounds and philosophies of the industries (telecommunications and the Internet respectively) in which they have their origins.

A shared requirement of Voice over IP protocols and applications is the need for each host engaged in a multimedia call or session to maintain a fixed address over the duration of the call. When either host is a portable computer on a wireless network, such as an 802.11 local-area network, mobility might result in the host needing to change IP address as it crosses subnet boundaries in the network.

Mobile Voice over IP (MVOIP), presented in this paper, provides a mechanism to maintain a VOIP call even as the underlying network addresses of the hosts engaged in the call need to change. Although it can be used by any real-time application, the current implementation of MVOIP is based on the ITU-T H.323 Multimedia standard, and we begin with a description of the H.323 protocol stack in the following section. We then describe the design and implementation of the MVOIP system in Sections 3 and 4 respectively, leading to a discussion, in Section 5, of experimental results. The sixth section discusses related work, and the seventh ends with conclusions and future work.

2. Background

The H.323 Multimedia Standard defines a *call* as a point-to-point multimedia communication between two or more H.323 *endpoints*, that begins with a call set-up procedure, and ends with a call termination procedure. An *endpoint* is an entity that can call and be called, and that generates and/or terminates information streams.

Figure 1 illustrates the five types of information streams (Audio, Video, Data, Communications Control and Call Control) that H.323 supports, and their related protocols.

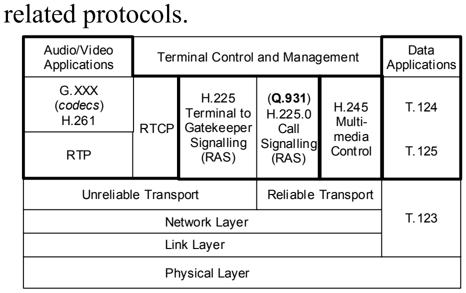


Figure 1: The H.323 protocol stack (from [11])

Message flow in a typical H.323 call begins with the exchange of Q.931 call-establishment messages. The H.245 communications control protocol is then used to exchange and negotiate capabilities, and to establish and open channels for the exchange of real-time data. Finally, the IETF Real Time Protocol (RTP) and Real Time Control Protocol (RTCP) [20] are used to transmit and receive the encoded audio/video stream. A single ongoing H.323 call thus consists of concurrent signaling, control and media channels open between the two communicating endpoints, as shown in Figure 2.

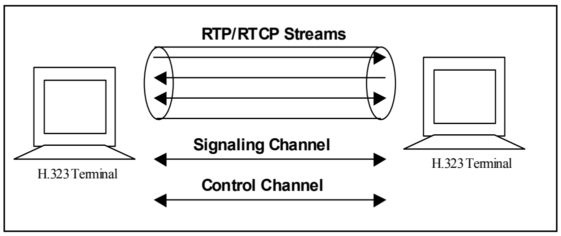


Figure 2: Communication streams in an H.323 call

Consider an H.323 endpoint on a wireless network such as an 802.11 wireless LAN. For the endpoint to take full advantage of the mobility afforded by the wireless network, the host should be able to physically roam to any point on the wireless network while still maintaining any ongoing calls. For such roaming to be supported, two levels of mobility support are necessary:

- *Micro-mobility*: The host must be able to seamlessly switch between wireless access points on the same subnet of the network.
- *Macro-mobility*: When the host moves into a new subnet, it must be able to detect this change and acquire a valid IP address for the new subnet. In addition, the change in IP address must not necessitate the termination of current VOIP calls.

Micro-mobility, as described above, is handled at the link layer by the 802.11 implementation. The macro-mobility scenario defines the problem to be solved by MVOIP. Handling this problem at the application level enables the application to respond in the most suitable way to the change in network address. In real-time applications, such as voice and video, that are sensitive to data transmission latency, a suitable response is to resume ongoing calls as quickly as possible. In addition, a mechanism must exist for future incoming calls to reach the mobile node at its new address.

3. MVOIP

Using MVOIP, hosts engaged in a call communicate about mobility using MVOIP messages, specifically the “Mobility Alert” and “IP Update” messages. These messages, which the mobile host sends to the non-mobile host when a network address change occurs, allow the communicating endpoints to directly hand off ongoing calls as soon as a need for such a handoff is observed, without the intervention of a third party, such as a mobility agent/server. This minimizes the hand-off latency for ongoing calls, an important requirement for real-time applications. When MVOIP is used in conjunction with a directory service (for address-resolution) such as that implemented in [7], updates to the directory service occur after the hand-off is complete and the call has been resumed, thus allowing the call to be resumed as quickly as possible.

3.1 Using MVOIP in call-handoff

The process of using MVOIP messages to hand off an ongoing call is outlined below, first from the perspective of the mobile host, and then from the perspective of the non-mobile host. In this discussion, “mobile” is used to refer to a host that is roaming across the network and is in the process of crossing a subnet boundary. The remote party with whom the mobile host is in a call is referred to as “non-mobile”. The classification of a host as “mobile” or “non-mobile” is dynamic over the course of a call, but we assume that the two communicating hosts do not simultaneously cross subnet boundaries (see Section 4.4).

From the perspective of the mobile host, using MVOIP in call hand-off involves several steps:

0. **Discover:** The mobile host determines that a hand-off is necessary, i.e., that it has crossed a subnet boundary. Then the mobile host initiates the “hand-off” process in the following steps.

1. **Mobility Alert Message:** When the mobile host determines that it is in a new subnet, it sends an MVOIP “Mobility Alert” message to the non-mobile host alerting it to the subnet change.

2. **IP Renew:** It then obtains a new IP address valid for the new subnet (e.g., from the DHCP server).

3. **IP Update Message:** Once it has a new IP address, it sends an “IP Update” message to the non-mobile host, reporting its new IP address.

4. **Resume:** Finally, using its new IP address, the mobile host updates its H.323 connections with the non-mobile host, allowing the call to continue.

From the perspective of the non-mobile host, handing off a call using MVOIP involves the following steps:

1. **Listen:** The host listens for Mobility Alerts or IP Updates from corresponding endpoints.
2. **Mobility Alert Message:** When the non-mobile host receives a Mobility Alert from a corresponding party, it pauses its call to that party.
3. **IP Update Message:** When it receives an IP update from the corresponding party, it updates its connections to that party, allowing the call to continue.

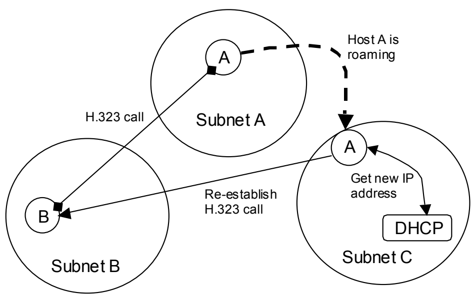


Figure 3: Changing subnets using MVOIP

3.2 MVOIP “Mobility Alert” and “IP Update” messages

The Mobility Alert message is a courteous notification to the remote party that a subnet change is occurring. The mobile host may send the Mobility Alert to prevent the call from being abandoned if the broken connection is detected before the IP Update message arrives. It can also be used as a prompt for the VOIP application on the non-mobile host to display a useful message to the user during the hand-off process. For implementation reasons discussed in Section 4, the Mobility Alert is a useful but optional message and its receipt by the non-mobile party is not essential to the correct functioning of the hand-off procedure.

The IP Update message is a notification of the mobile host’s new IP address after a subnet change. It prompts the non-mobile host to continue its communication with the mobile host at the new IP address.

MVOIP messages have the following fields:

<Version>
An MVOIP version number, currently 1.0.

<Type>
Message type: 0 for a Mobility Alert and 1 for an IP Update.

<Call ID>
The identifier for the call or multimedia session in question.

<Source>
The old IP address of the mobile host that is the sender of the Mobility Alert or IP Update.

<New IP>
Null for Mobility Alerts. For IP Updates, it is the new IP address of the mobile host (the sender).

<Destination>
The IP address of the recipient, i.e., the non-mobile host.

<Timestamp>
The time in seconds since the beginning of the call, at the instant when the mobile host sends the message.

<Update Number>
Null for Mobility Alerts. For IP Updates, it is the number of IP Updates that have been sent so far, starting at 1 for the first subnet change.

3.3 Security in MVOIP

There are several potential security concerns in MVOIP due to a malicious third party intercepting and modifying MVOIP messages, or even generating and sending fake MVOIP messages to the non-mobile host. The following scheme for security in MVOIP depends on the existence of a shared secret or key between the two hosts in the call. This key can be established via an authenticated directory (e.g., see [7]) or key-distribution service, and is used to encrypt the MVOIP messages.

The <Timestamp> and <Update Number> fields are used to enforce security in MVOIP. The <Timestamp> is the length of time in seconds since the beginning of the call in question. The <Timestamp> field is included in every Mobility Alert or IP Update message that a mobile host sends out. The <Update number> is set to null for Mobility Alert messages. For IP Update messages, it is the number of IP Update messages, including the current one, that mobile host has sent to the non-mobile host since the beginning of the current call. It starts with a value of 1 for the first subnet change, and increments by 1 for every subsequent IP Update message. The mobile host encrypts all MVOIP messages using the shared secret key before sending them over the network.

When the non-mobile host receives a Mobility Alert message, it decrypts it using the secret key for the current active call, and checks the timestamp field. If the timestamp is off by more than a few seconds, or if an IP Update message has been received from the mobile node in the last few seconds, the Mobility Alert message is discarded. Checking the timestamp prevents replay attacks in which a third party can capture a Mobility Alert message sent by the mobile host, and re-send it later in an attempt to disrupt the call by causing the non-mobile node to pause the call. If the attacker resends the message as soon as it is captured, i.e., when the mobile node has not yet sent an IP Update message, the replay attack has no adverse effect on the call, and the non-mobile host can simply ignore the duplicate Mobility Alert message. Note that because all messages are encrypted, an attacker cannot compromise a call by constructing and sending a fake Mobility Alert message.

When the non-mobile host receives an IP Update message, it decrypts it using the secret key, and checks the <Update number> field against the expected update number. If the numbers are not equal, it discards the IP Update because it is indicative of a replay attack. Encryption prevents the attacker from sending fake IP Update messages and from intercepting and modifying genuine IP Update messages.

3.4 Calling into Mobile Hosts

A host wishing to call another host must be able to discover the IP address of the host to be called. Because hosts are mobile and can change IP addresses at any time, MVOIP needs to be used in conjunction with a dynamic directory infrastructure that will always contain the most current IP address of the hosts using the service. Using this directory system, a mobile host can always be reached with incoming calls. The directory infrastructure implemented by Ammar Khalid [7] meets MVOIP's needs in this regard.

4. Implementation of MVOIP

The MVOIP protocol is platform independent and can be implemented for any Voice over IP application. The current implementation of MVOIP is based on the OpenH323 project's open-source implementation of the H.323 standard [2]. Our implementation works on a DHCP-enabled Windows 2000 platform that, to be a mobile host, must be equipped with a Lucent "Gold" 802.11-compliant network interface card and must be within range of any 802.11-compliant wireless network that supports the Dynamic Host Configuration Protocol (DHCP). Any Windows 2000 platform connected to the Internet and running the MVOIP software can be a non-mobile MVOIP host. The current implementation does not include the security measures described in Section 3.3 above.

4.1 Identifying subnet changes ("Step 0" on the mobile host's end)

The mobile host needs to know when it has switched subnets in order to initiate the hand-off procedure. A host on an 802.11 wireless network is in constant communication with at least one wireless access point, and a mobile host's first indication of a possible subnet change is the discovery that it is associated with a new access point. A subnet may contain several access points. In the current implementations of 802.11 (Lucent and Cisco implementations), a client can directly query its associated access point for the access point's hardware MAC address, but not its IP addresses or subnet number. The mobile host must thus use additional methods to determine when it crosses a subnet boundary.

Our implementation uses one or more "hints" from the network to determine when a subnet change occurs, and the effectiveness of each of these hints is evaluated in Section 5. The first hint, as already mentioned, is the discovery that the mobile host is associated with a new access point. The host polls the network interface card driver to determine if an access point change has occurred. One of two additional hints is used to determine subnet changes:

Time elapsed since receipt of last RTP packet: If the non-mobile host has *silence detection*[1] disabled, then the mobile host will receive packets of audio data at regular time intervals over the duration of the call. In our case, using the GSM 06.10 codec, the receive time interval is approximately 80ms. If the host encounters a new access point and has not received an RTP packet from the remote party within a few receive intervals, it assumes that it has changed subnets. As shown in the Experiments and Results section, this hint is less effective when the remote host enables silence detection in its audio codec and is in a normal conversation in which it is silent for long periods of time. We diminished this problem by having the remote party send periodic "heartbeat" RTP packets when silence detection is enabled.

Ping time to fixed IP address: To use this hint, when the VOIP application starts up, the mobile host "pings" a fixed IP address a few times to determine

[1] When s*ilence detection* is on, no RTP packets are sent during silences in the voice stream, to conserve network bandwidth.

the average round-trip time. When it detects an access-point change, it attempts again to ping this IP address, and if it does not receive a reply within a reasonable time, determined by the average time measured above, it assumes that its own IP address is no longer valid and that it has changed subnets. Because any IP-enabled host on a network connected to the Internet has a default gateway that is guaranteed to be one hop away on the same subnet, the default gateway's IP address is an ideal address to ping for this hint.

After the mobile host, using one or more of the above hints, determines that it has changed subnets, it initiates the hand-off procedure.

4.2 MVOIP "Mobility Alert" and "IP Update" Messages

The first step in the hand-off procedure is for the mobile host to send a "Mobility Alert" message, as a UDP packet, to the non-mobile host. UDP is used because by the time a subnet change is detected, the IP address of the mobile host is no longer valid and so a TCP connection cannot be established. Because UDP does not guarantee packet delivery, and also because ingress filters in network routers may discard packets originating from invalid IP addresses, the implementation of the hand-off procedure does not depend on the receipt of the Mobility Alert message. Still, if the message does arrive, it can be a helpful hint to the non-mobile host.

The mobile hosts sends the "IP Update" message via TCP because the receipt of this message is essential for the correct continuation of the call, and by the time this message is sent, the mobile host has a valid IP address for the new subnet.

4.3 Handing off the H.323 Call

To hand off an H.323 call, each of the channels of information flow (the signaling, control and media channels) are paused, reset or reconfigured where necessary, and un-paused. For TCP channels (such as the call signaling channel), this involves establishing a new TCP connection using the updated IP address. For UDP channels (used for the media stream), it is sufficient to reset the sockets used for communication and update the metadata concerning the remote party. Although new network connections need to be established between the communicating parties for each channel of information flow between them, at no point during the hand-off are the abstract communication channels between the parties shut down or terminated. As such, to re-establish the call, the parties do not need to repeat the H.323 call signaling process, nor do they need to re-negotiate capabilities or negotiate the opening of any logical channels, because these capabilities are already in effect and are remembered, along with the call reference number, during and after the hand-off process. The latency of the hand-off procedure is hereby kept to a minimum.

4.4 Handling Error Situations

A failure to obtain a new IP address, or a situation in which both parties change subnets simultaneously, making it impossible for them to re-establish their connections since they now do not know how to contact each other, will result in a failure of the hand-off process. In this case, the call is ended and the hosts might choose to try contacting each other again with a new H.323 call. In the case when both hosts simultaneously cross subnet boundaries and as such no longer know each other's addresses, a dynamic MVOIP directory system for a host to look up another's current IP address is essential. Ammar Khalid [7] has implemented such a system.

5. Experiments and Results

Our experiments were aimed at examining the hand-off off latency and the factors that affect it, and also at evaluating the effectiveness of various methods of detecting subnet changes.

To measure the hand-off latency of MVOIP, we measured the number of RTP packets sent and received over the duration of a 1-minute long call that included one subnet change. Figure 4 plots the number of RTP packets of data sent and received per second by the mobile host. For this experiment, we used a laptop as the mobile host roaming from one subnet A to another B, in a call with a non-mobile host stationed in subnet B. Both hosts used a Lucent "Gold" WaveLAN card to communicate at 11Mbps to access points connected to the 10Mbps Ethernet network, ran our modified version of the OpenH323 software on Windows 2000, and had the silence-detection option turned off. In the figure, the handoff process is observed as a 3.3-second long period during which no packets are sent or received by the mobile party. The spike in the number of packets sent as soon as hand-off is completed is due to buffering of data by the audio codec.

For the call illustrated in Figure 5, a breakdown of the handoff duration into its component steps (i.e., sending the Mobility Alert, pausing the call, releasing the old IP address, obtaining a new IP address, sending the IP Update, and finally resuming the call) shows that the majority of the hand-off time is spent in the Windows 2000 API call to obtain an IP address from the DHCP server. This is because the operating system verifies, by broadcasting several *Address Resolution Protocol (ARP)* messages, that the returned IP address is not in use by another host on the subnet, before returning from the API call.

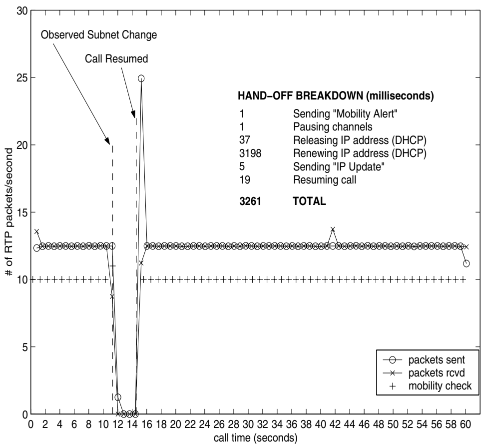


Figure 4: Packets sent and received by mobile host during a call with a subnet change

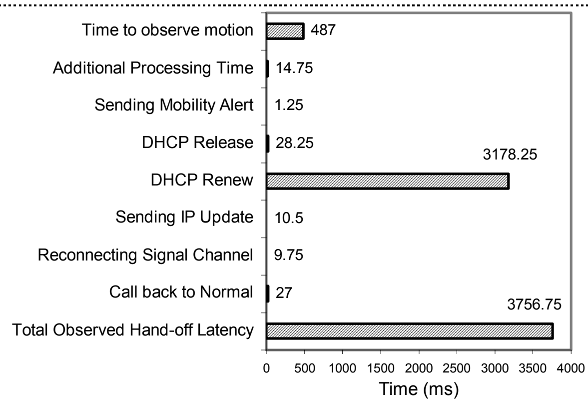


Figure 5: Breakdown of hand-off latency

Figure 5 shows a breakdown of the handoff latency averaged over 4 calls. The "Time to observe motion" is the average amount of time between when the mobile host switches subnets, and when it actually observes the change. It depends on the frequency at which the application polls the wireless card driver to detect new access point associations. In these test runs, the polling frequency is once per second. Increasing the polling frequency to 5 times a second reduces the average time to observe an access point change from 487ms to 177ms.

Due primarily to the amount of time it takes to perform the DCHP IP address renewal, there is a perceivable period of silence during call hand-off. Hand-off takes approximately the same length of time whenever it is initiated, regardless of whether a subnet change actually occurred. For this reason, it is important that the mobile host initiates the hand-off process only when it is absolutely necessary to do so, that is, when it crosses a subnet boundary. As discussed in section 4.1, our implementation of MVOIP uses hints to determine when the mobile host undergoes a subnet change, and Figure 7 compares the effectiveness of these hints. In the test scenario that yielded these results, the mobile host roamed back and forth five times across three subnets, thus undergoing a total of twenty subnet changes (see Figure 6). The number of access-point changes the mobile host makes depends on which access points it associates with as it roams, which in turn depends on perceived signal strengths and other factors built into the 802.11 implementation. We have no direct control of these associations, beyond ensuring that mobile host follows the same physical path in each test run.

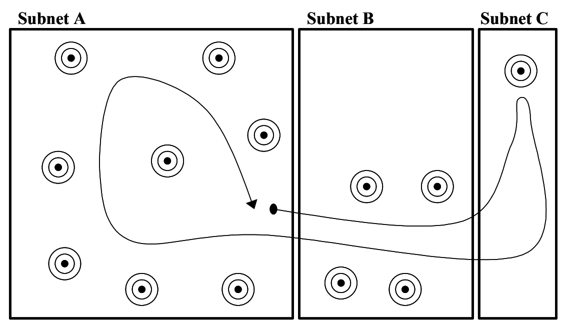
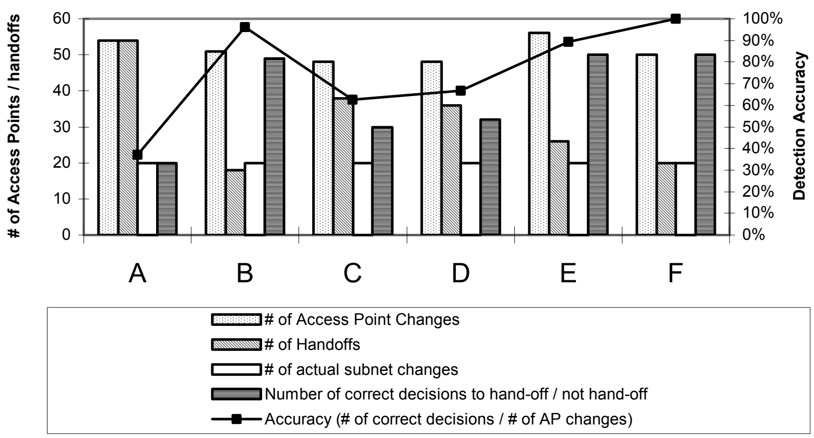


Figure 6: Test Scenario for the results in Figure 7. The mobile follows the indicated path across 3 subnets five times, thus undergoing 20 subnet changes and associating with whichever access points are necessary as it roams.

Figure 7 indicates that, of the hints we implemented, pinging a known IP address (F) such as the default gateway is the most effective way to determine when a subnet change occurs. Unlike the hint that uses RTP receive times in conjunction with RTP "heartbeat" packets (D and E), it requires no change to the RTP implementation, and is accurate 100% of the time. Using ping hints however adds the ping timeout period (averaging 250ms in these tests) to the call hand-off time, bringing the average time for call handoff to up to 4.145 seconds.

The hand-off latency is a bottleneck because it results in a long period of silence during which the mobile host loses packets of data. This silent period could be reduced if the mobile host could immediately begin using the IP address returned from the DHCP server, while verifying in the background that it is not in use by any other host.

Alternatively, if the mobile host could still receive packets destined to its old IP address while it is going through the hand-off process, the hand-off duration would be less critical. Implementing such a “soft” rather than a “hard” hand-off between 802.11 wireless access points would make it possible for the mobile host to be associated with two access points simultaneously during the MVOIP hand-off process. This approach would require modifications to the 802.11 WLAN specification.



The hints used are the following:
A) Access point changes only. **B)** Time since last RTP packet received, in conversation with no silence detection. **C)** Time since last packet received in conversation with silence detection **D)** Time since last packet received in conversation with silence detection and RTP “heartbeats” sent every 500ms **E)** Time since last packet received in conversation with silence detection and RTP “heartbeats” sent every 100ms **F)** Ping to default gateway only.

6. Related Work

Several schemes have been proposed to handle mobility in IP networks and voice applications.

The IETF’s Mobile IP [14] proposes to handle mobility at the IP level and it hides the movement of the mobile host from the upper layer protocols and applications by using *Home Agents* (HA) and *Foreign Agents* (FA) to handle the routing of IP packets to the mobile host. The network where a host begins is its *home network*. When the host moves to a foreign network, it obtains a *care-of-address* (COA) via registration with a Foreign Agent, and it registers with its Home Agent to forward all in-coming packets to the COA using IP-IP encapsulation or “tunneling” (see Figure 8). In Route Optimized Mobile IP [15], the remote node with which the mobile node is corresponding (called the *Corresponding Node*) is informed of the current care-of-address of the mobile node to avoid triangle routing caused by forwarding packets through the Home Agent.

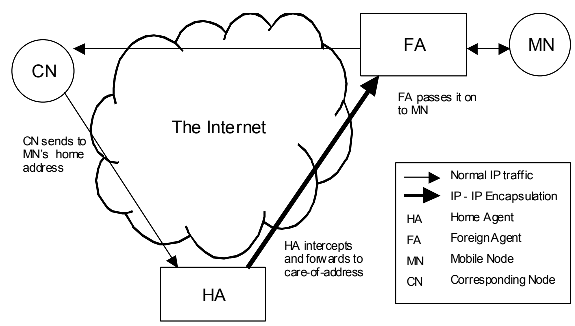


Figure 8: IP packet forwarding in Mobile IP

Mobile IPv6 [16], designed to work with the next generation of the Internet Protocol, IPv6, requires all corresponding nodes and intermediate network routers to support caching of these *binding updates*, thus reducing triangular routing to a minimum. The advantage of Mobile IP is that mobility is completely transparent to higher-level protocols, and with the ubiquitous deployment of IPv6, it promises to be the solution to IP mobility in future networks. It is not yet supported by many hosts or routers.

Other proposed IP layer mobility support systems include HAWAII, proposed by Ramjee et al [17] [18], and a multicast based system proposed by Helmy [6]. Both these systems are geared towards low-latency real-time data and avoid the triangular routing found in Mobile IP. They do, however, require special IP-layer support in the form of sophisticated base stations in the former and multicast support in the latter.

Liao [8] proposes an application-level protocol, Mobile Internet Telephony Protocol (MITP), to handle mobility in Internet telephony applications. In MITP, the mobile host handles the hand-off of the ongoing call by the join and departure of a multi-point conference. For example, consider a Host A with address "a" in a conversation with Host B with address "b". Suppose the call-id for this call is "ab". Host A then moves to a new subnet where it now has an address "a2". It contacts Host B at address "b", requesting to join the call with id "ab" using address "a2". After it has joined the call, it sends another message to Host B, removing "a" from the call. MITP has not been implemented [9].

Wedlund and Schulzrinne [22] propose a mobility scheme using the Session Initiation Protocol, (SIP), an application-level protocol for establishing and ending multimedia sessions, and an alternative to H.323. In SIP, a multimedia session is initiated with an INVITE message containing a session description including *call-id*, *to* (the originator of the message), *contact* (again, the originator of the message), and *from* (the recipient of the message) fields. After a subnet change, they propose to implement hand-off by sending a re-INVITE message to the corresponding node. This is simply an INVITE message where the *to* field contains the "home" address of the mobile node, and the *contact* field contains its new address (obtained by a method such as DHCP). In this way, the multimedia session is re-established between the two nodes. They propose to handle long-lived TCP connections by a means such as Mobile-IP. The advantage of this scheme using SIP is that it is simple and makes use of an existing SIP message. Because of this though, it can only be used with SIP-initiated calls, which precludes its use with H.323. The MVOIP messages, on the other hand, are essentially a means of exchanging mobility information, and could be used in any real-time application, including those using SIP. In addition, the implementation lessons on subnet detection accuracy and DHCP latency, learned from MVOIP, are relevant to other hand-off schemes such as this one.

Park et al [12] propose a hybrid application-level and IP-level scheme for handoff management in H.323 calls. Their system proposes to maintain the signaling (Q.931/H.225) and control (H.245) channels during hand-off using Mobile IP as these channels are not sensitive to communication latency. For the media stream, they propose to handle hand-off by simply closing the old logical channel and opening a new logical channel directly between the two communicating parties using the Mobile-IP care-of-address. The call-control functions already built into the H.245 protocol will enable the opening and closing of logical channels. This scheme has not been implemented, but its advantage lies in its use of existing H.323 protocols. It does, however, require Mobile IP, and handoff includes the additional latency of closing and opening H.323 logical channels.

Finally, several parties, including Motorola [10] and AT&T [19] have made proposals for ITU's extension for mobility support, referred to as H.323 annex H. These contributions are high-level descriptions of required mobility support across all aspects of the H.323 standard, concentrating especially on Gatekeeper[2] discovery and registration. In terms of handing off an ongoing call, which is the main focus of our work, Motorola specifies a hand-off process from one *Wireless Access Unit (WAU)* to another, in which a temporary channel is established between the old and the new WAUs for packet forwarding until the mobile host is fully associated with the new WAU. The Wireless Access Unit is a functional entity that houses and manages the radio transceivers and handles the radio-link protocols with the Mobile Host. It appears to have a similar function to the link-layer 802.11 wireless access point, but has extended capabilities geared specifically to H.323. They do not specify how the hand-off process handles changing IP addresses. AT&T, on the other hand, briefly suggests a hand-off process similar to Liao's MITP, that involves the join and departure of a multi-point conference. We believe that MVOIP could easily be plugged into either of these high-level proposals, to handle the problem of handoff in an ongoing call.

While the long-term solution to IP mobility may lie in systems that make mobility transparent to higher level protocols, such as Mobile IP or Helmy's multicast system, the infrastructure requirements of these schemes reduce the feasibility of their current deployment. Another challenge in handling mobility transparently to higher-level protocols is the conflicting requirements of these protocols. While

[2] An H.323 *Gatekeeper*'s functions include address translation, authentication of terminals and gateways, bandwidth management, accounting, and billing.

real-time applications such as voice are tolerant to data loss but intolerant to latency, other applications require loss-less delivery of data and are less stringent in their latency requirements. For these reasons, we believe that an application-level scheme such as MVOIP is a good approach to support mobility in current VOIP applications.

MVOIP can be extended for generalized application-level mobility support by providing subnet detection and automatic acquisition of new IP addresses in the operating system, or as a background daemon. The daemon would inform interested applications of the changes, allowing them to respond in an appropriate manner for the application, which would include exchanging MVOIP messages to keep ongoing sessions alive.

7. Conclusions

The duration of the hand-off period would be significantly reduced if the mobile host could immediately begin using the new IP address while verifying in the background that it is not in use by any other host. In addition, the accuracy and speed of determining when handoff is necessary could be increased if the 802.11 network interface card could notify the application when it associated with a new AP, to prevent continuous polling of the driver, and also if the access points could be queried for their subnet numbers.

Despite the DHCP bottleneck, MVOIP is a feasible means of handling mobility that can be integrated with real-time applications and requires no special-purpose hardware or support at the IP level. Because hand-off in MVOIP is strictly between the two communicating clients and does not require the use of specialized agents or servers other than the DHCP server, it is a scalable approach to mobility.

MVOIP can be extended to handle situations where the network connection point changes without necessarily undergoing a subnet change, e.g., when a user switches from an Ethernet to an 802.11 wireless connection. MVOIP should be extended to support the remaining components of the H.323 standard, most importantly, multipoint conferences and use of the Gatekeeper.

This project was supported by the Cisco University Research Program, by Lucent Technologies, by Microsoft Research, and by DoD DURIP contract number F49620-00-1-0212.

8. References

[1] ITU-T Recommendation H.323 (1998), *Packet-based multimedia communication systems.*

[2] The OpenH323 Project, http://www.openh323.org.

[3] B. Douskalis, *IP Telephony: The Integration of Robust VOIP Services*, Prentice Hall, 2000.

[4] A. Dutta, F. Vakil, J.-C. Chen, M. Tauil, S. Baba, N. Nakajima, H. Schulzrinne, "Application Layer Mobility Management Scheme for Wireless Internet", *3G Wireless* 2001, May 2001, pp. 7.

[5] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, "SIP: Session Initiation Protocol", RFC 2543, Mar. 1999.

[6] A.Helmy, "A multicast-based protocol for IP mobility support," *Proceedings of NGC 2000 on Networked group communication*, Nov. 2000, Stanford, pp. 49-58.

[7] A. Khalid, "A Directory Infrastructure to Support Mobile Services," Computer Science Technical Report TR2001-391, Dartmouth College, May 2001.

[8] W. Liao, "Mobile Internet telephony protocol: an application layer protocol for mobile Internet telephony services," *IEEE International Conference on Communications ICC '99*, 1999 Vol.1, pp: 339 –343.

[9] W. Liao, e-mail communication, 16 May 2001.

[10] E. Martinez, Motorola, "Enhancements to ITU-T Recommendation H.323 to support User and Service Mobility," ITU Document APC-1646, Oct. 1999.

[11] M. Miller, *Voice over IP: Strategies for the converged Network*, M&T Books, 2000.

[12] D. Park, W. Yoon, and D. Lee, "An Efficient Handoff Management for Mobility Support to H.323," *International Workshop on Mobile Multimedia Communications*, Oct. 2000.

[13] B. O'Hara and A. Petrick, *IEEE 802.11 Handbook: A Designer's Companion*, IEEE Press, 1999.

[14] C. Perkins, "IP Mobility Support," RFC 2002, Oct. 1996.

[15] C. Perkins and D. Johnson, "Route Optimization in Mobile IP," IETF Internet draft, Feb. 2000.

[16] C. Perkins and D. Johnson, "Mobility support in Ipv6", *Proceedings of the Second Annual International Conference on Mobile Computing and Networking*, Nov. 11-12, 1996, pp. 27-37.

[17] R. Ramjee, T. La Porta, L. Salgarelli, S. Thuel and K. Varadhan, "IP-Based Access Network Infrastructure for Next-Generation Wireless Data Networks," *IEEE Personal Communications*, Aug. 2000, pp. 34-41.

[18] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, L. Salgarelli, "IP micro-mobility support using HAWAII," IETF Internet draft submission, Jul. 2000.

[19] R. Roy, AT&T, "H.323 Mobility Architecture and Protocol for Terminal, User and Service Mobility," ITU Document APC-1652, Oct. 1999.

[20] H. Shulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", RFC 1889, Jan. 1996.

[21] H. Schulzrinne, E. Wedlund, "Application-Layer Mobility Using SIP", *Mobile Computing and Communications Review*, Vol. 4, No. 3, Jul. 2000, pp. 47-57.

[22] E. Wedlund, H. Schulzrinne, "Mobility Support using SIP", ACM/IEEE International Conference on Wireless and Multimedia (WOWMOM), Aug. 1999, pp. 76-82.